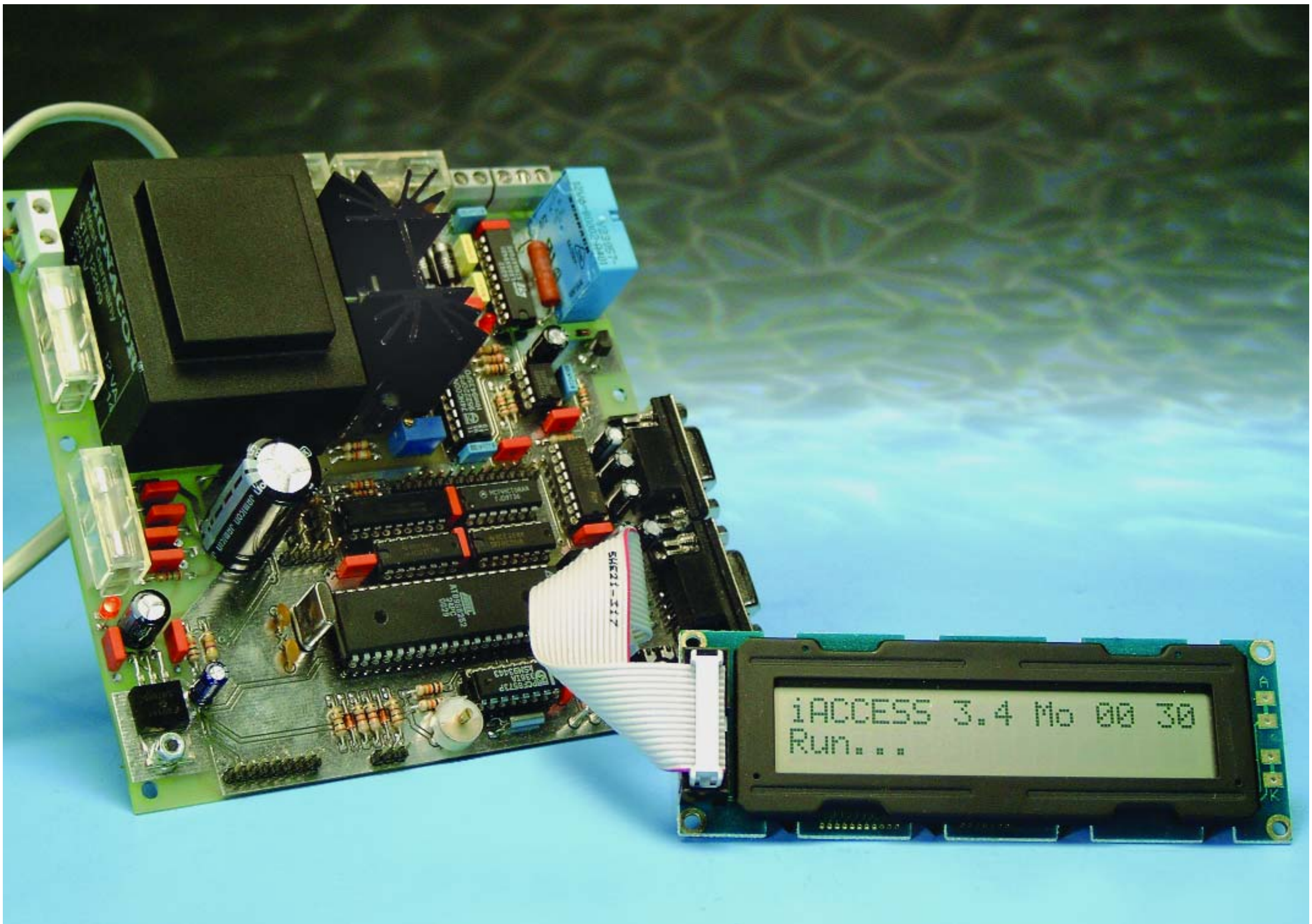
iAccess

an intelligent access control system

Design by I. Joostens

Surveys show that people are feeling increasingly less safe. Consequently, protecting property and buildings receives more and more attention these days. One of the aspects of such protection is access control. The system described here goes further than most such systems. Besides granting persons selective access to specific areas, iAccess allows you to check who was granted access to which area and when.



Features at a glance

- up to 125 keys can be programmed
- independently controls two entrances using two readers
- access privileges can be individually configured for each key
- protected against scanning
- system configuration using an LCD menu or an RS232 link to a PC
- outputs can be configured as monostable (door opener) or bistable (relay)
- door openers can be powered by AC or DC voltages
- optional indicator LEDs
- programming port for easy firmware updating
- activities can be logged via an RS232 link
- backup power supply using a lead-acid battery
- high reliability ensured by a watchdog timer

Nowadays, access control systems are available from quite a few manufacturers. These are usually either quite elaborate systems designed for thousands of users and offering a large number of features, or very limited models that are more suitable for private use.

Most commercially available systems use electronic keys, such as magnetic cards, badges, iButtons, or transponders. There are also models that use PIN codes and/or biometric features.

The circuit described in this article combines the versatility and features of the large systems with the simplicity and ease of use of the small, stand-alone types. Here iButtons are used as electronic keys. Relative to other key systems, iButtons offer quite a few advantages, such as ready availability, a high level of robustness, and the fact that the user interface (reader) consists of only a simple set of contacts. **Figure 1** shows a photo of an iButton with its associated read probe.

Two entrances are independently monitored. The circuit includes a driver for a door opener as well as a relay output. The latter is suitable for enabling or disabling an alarm system, among other things. An optional link to a computer can be used to configure the system and record events in a log file.

Concept and block diagram

The overall structure of the access control system is shown in **Figure 2**. As can be clearly seen, the key element is a microcontroller. It looks after reading the iButtons, handles communications via the RS232 interface and controls the peripheral components.

The microcontroller used here has 2 KB of EEPROM, which stores the physical key numbers (as engraved on the iButtons) and associated access privileges, as well as the system configuration parameters.

A special programming interface allows the microcontroller software to be easily updated. As the program uses a fairly large number of interrupts, it was not possible to have the microcontroller maintain the time of day using software. Consequently, a real-time-clock IC is used for this purpose.

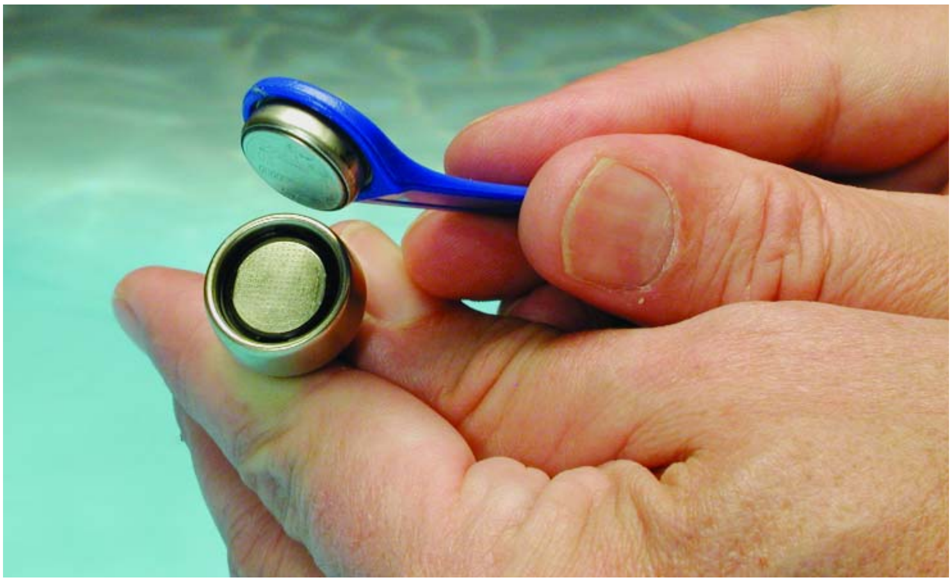


Figure 1. A DS9092 iButton with associated read probe.

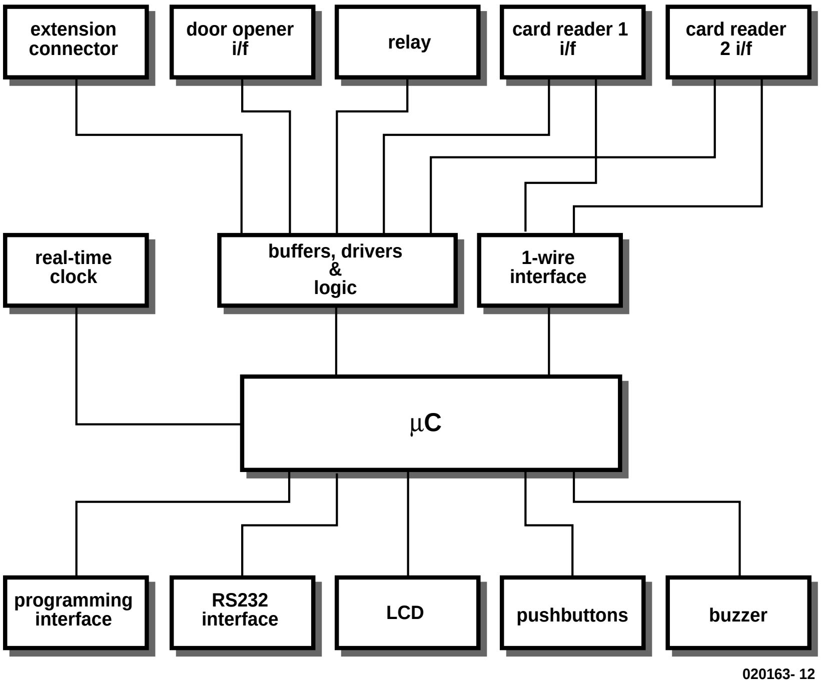


Figure 2. Block diagram of the access control system.

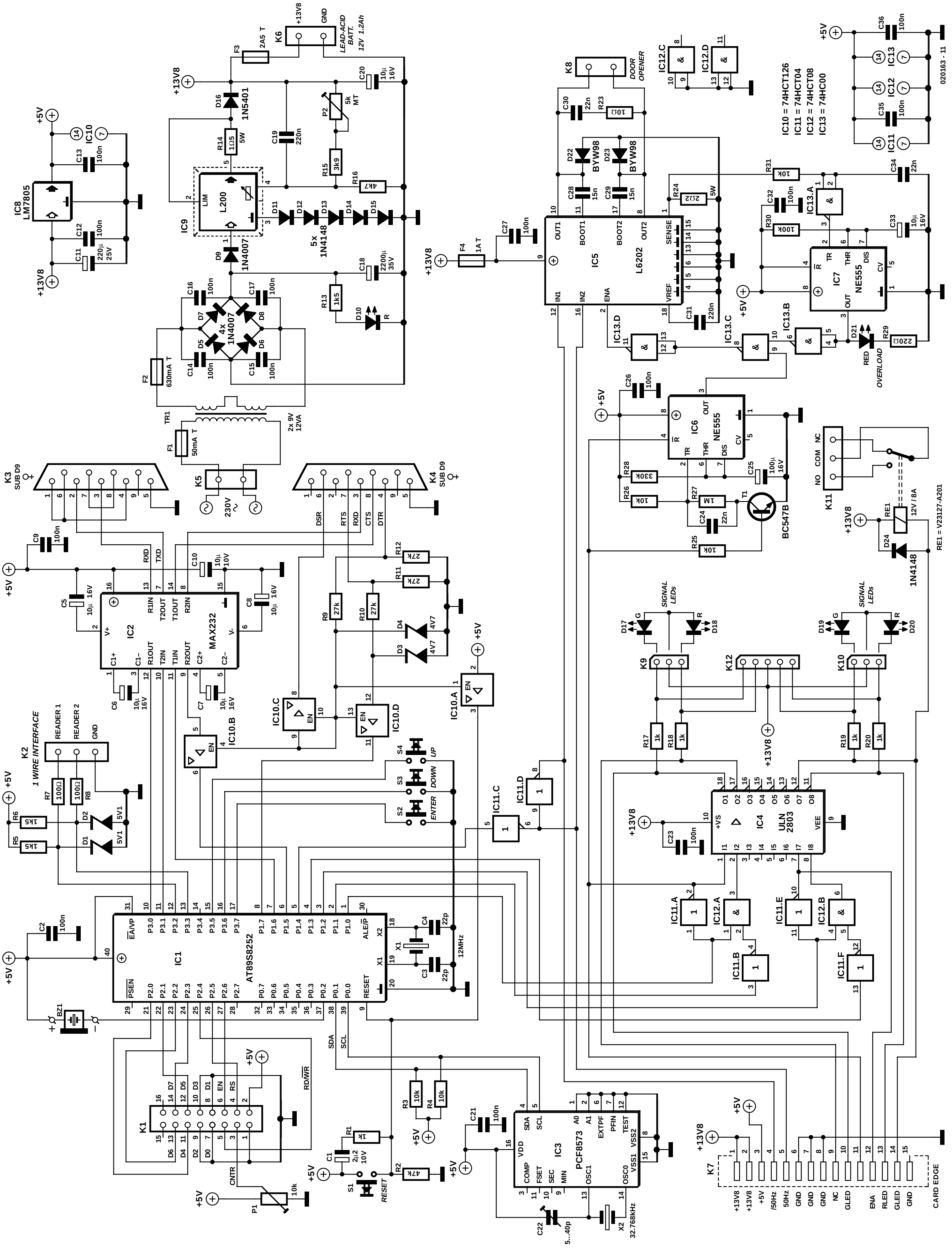


Figure 3. Schematic diagram of the complete system.

The user interface is formed around an LCD module, four push-buttons (including the reset button) and a miniature buzzer. To reduce the number of I/O pens used, the LCD is driven in 4-bit mode.

The iButtons are read using an intentionally simple 1-wire interface. As the length of the 1-wire bus will be relatively short in most cases, and only one 1-wire device will be connected to the bus each time, the bus does not have to meet particularly severe requirements. Key readers can be fitted with status indicators using red and green LEDs.

The system can drive most types of door openers that are suitable for use with a 12-V supply voltage. A circuit specially developed for this purpose allows the door opener to be driven using an AC or DC voltage as desired. To prevent possible damage to the access control system or the door opener in case of improper use or fault conditions, several supplementary protective measures have been incorporated.

As already mentioned, besides the door opener driver circuit there is also a relay output that can be used for additional switching functions. Finally, an expansion connector has been added for users who wish to experiment with the circuit.

Power for the entire circuit is provided by a lead-acid storage battery that is continuously recharged by a mains power supply. This provides two significant benefits. First, in case of a power outage the system will continue to operate from the battery for an extended length of time, so access to an area or building remains assured. Besides this, using a storage battery allows the mains power supply to be relatively modest in size, since the short current pulses needed to operate the door opener are provided by the battery instead of the mains supply.

Schematic diagram

The full schematic diagram of the system is shown in **Figure 3**. The functional blocks shown in Figure 2 can be recognised here without too much effort. The most important elements are described in more detail below.

Microcontroller, user interface, RS232 interface and programming port

Microcontroller IC1, an AT89S8252, is used here in a quite conventional manner. No external program memory is used; all of the firmware is located in the microcontroller's 8-KB Flash memory. A fully programmed version of the microcontroller can be purchased from Readers Services, but the source code and executable software are also available from the *Elektor Electronics* website or on a pair of diskettes (see the Components List).

The microcontroller can be programmed via connector K4 using a standard 1:1 serial cable (not a null-modem cable) and a computer running the *Elektor Electronics* MicroFlash program. The microcontroller must be held in the reset state while it is being programmed. This is done using pin 4 of K4 (DTR), with the necessary level adjustment being provided by R9, R12 and D4, in combination with IC10, a 74HC126 quad tri-state buffer.

During programming, IC10 causes the signals on pins P1.5–P1.7 of the microcontroller (SPI bus) to be connected to K4. Here IC2 (MAX232) acts as a combined level adapter and RS232 driver. When the DTR line on pin 4 of K4 again goes low, the outputs of IC10 go into a high-impedance state, which isolates the programming interface from the rest of the circuit. In this state, S1 acts as a normal reset button (in combination with R1, R2 and C1).

The serial port (RS232) is built around an MAX232 in a conventional manner and is available on connector K3. BZ1 is a miniature piezoelectric buzzer that can be driven directly from a microcontroller I/O output.

Real-time clock

The real-time clock is formed by a PCF8573 (IC3) in combination with a watch crystal (X2) and a trimmer capacitor (C22). The IC, which can be completely controlled via an I_C interface, accurately maintains the date (day and month) and time of day (hours and minutes). R3 and R4 are included because the microcontroller P0 port does not have internal pull-up resistors.

1-wire interface

The 1-wire interface is connected to microcontroller interrupt pins INT0 (P3.2) and INT1 (P3.3) via several resistors and Zener diodes. When an iButton is held against a key reader, the associated I/O pin of the microcontroller is briefly pulled to ground. This generates an interrupt, which triggers an interrupt routine that ultimately reads the data from the iButton for further processing by the main program.

As the internal pull-up resistors in the microcontroller have fairly high resistances, R5 and R5 are included to provide additional current for the 1-wire bus. According to the 1-wire protocol specification, R5 and R6 normally have a value of at least 1.5 k_, but if the distance between the key reader and the access control system is relatively long, better performance can be obtained by using a somewhat lower value. From experimental tests, a value of 820 Ω will allow the system to easily handle a distance of more than 100 metres between the access control system and the key reader.

R7, R8, D1 and D2 protect the I/O pins against high voltages from electrostatic discharge or attempted sabotage. For instance, they enable the microcontroller to survive someone holding a 9-V battery against the contacts of a key reader.

Buffers, logic and drivers

A ULN2803 (IC4) is used as a driver IC for the various indicator LEDs and the output relay. This IC contains eight open-collector drivers, each of which can switch a maximum current of around 500 mA.

IC11 and IC12 are included because the microcontroller I/O pins cannot supply enough current in the high state to properly drive a ULN2803, and because we prefer to use active-low signals at the microcontroller level (since all I/O pins are high during a reset). These ICs have a dual role. First, they act as buffers between the microcontroller and IC4, and second, they form a logic circuit that also provides the necessary signal inversion. This allows the microcontroller software to be simplified. 50-Hz signals are present on pins 6 and 8 of IC11 for driving a door opener.

Door opener interface

The door opener is driven directly by IC5, an L6202. This IC, which is actually intended to be used as a motor driver, contains a full H-bridge using MOSFET technology. It can handle a continuous current of 1.5 A and peak currents up to 5 A at a maximum voltage of 48 V. Both halves of the H bridge are driven by the 50-Hz signals from the microcontroller. Whether the door opener actually receives any current depends on the signal at the

Enable input of the IC. This signal is set high by the microcontroller after a valid key has been presented. However, this does not happen just like that; there is also protective circuitry in the signal path.

The first protective circuit is built around IC6, which is a 555 wired as a monostable timer with a period of approximately 30 seconds. When the signal on pin 2 of IC11 goes high, T1 is switched on via R25. This causes the trigger input of IC6 (pin 2) to be briefly pulled to ground until C24 has been charged. The rising edge on pin 2 of IC11 does not affect the Reset input of IC6, which is active low, so the timeout starts running at this point. If the signal on pin 2 of IC11 returns to the low level during the 30-second interval, the falling edge on pin 4 of IC6 will reset IC6.

The direct effect of this is that the output signal on pin 3 exactly follows the Enable signal as long as it does not stay high longer than 30 seconds; otherwise IC6 will independently disable the output signal. This prevents the door opener from being enabled for an extended length of time, which would be detrimental to both the coil of the door opener and the lead-acid battery.

A second protective circuit prevents IC5 from being overloaded and acts as a sort of current limiter. The current flowing though the door opener and IC5 reaches ground via sense resistor R24. The voltage across R24 is fed via R31/C34 (for noise suppression) to NAND gate IC13a, which is wired as an inverter. As soon as the voltage on pins 1 and 2 of IC13 reaches half the supply voltage level, IC13 will see this as a ‘high’ logic level and pin 3 will go low. This occurs with a current of approximately 1.14 A (2.5 V ÷ 2.2 Ω). This will cause timer IC7 to be triggered via pin 2 and start a timeout of around 1 second. Pin 3 of IC7 will go high, and overload indicator D21 will be illuminated. The Enable signal on pin 2 of IC5 will be pulled low via IC13b–d, interrupting the current. After 1 second, IC7 will reset itself, and if the overload condition has been corrected the circuit will again be operational. If the overload is still present, IC7 will immediately be retriggered after resetting itself. The net result is a series of short current pulses at a one-second rate, which will not harm IC5.

Power supply

The power supply consists of a charging circuit built around voltage regulator IC9 and a small lead-acid storage battery rated at 12 V/1.2 A. To keep the ripple component of bat-

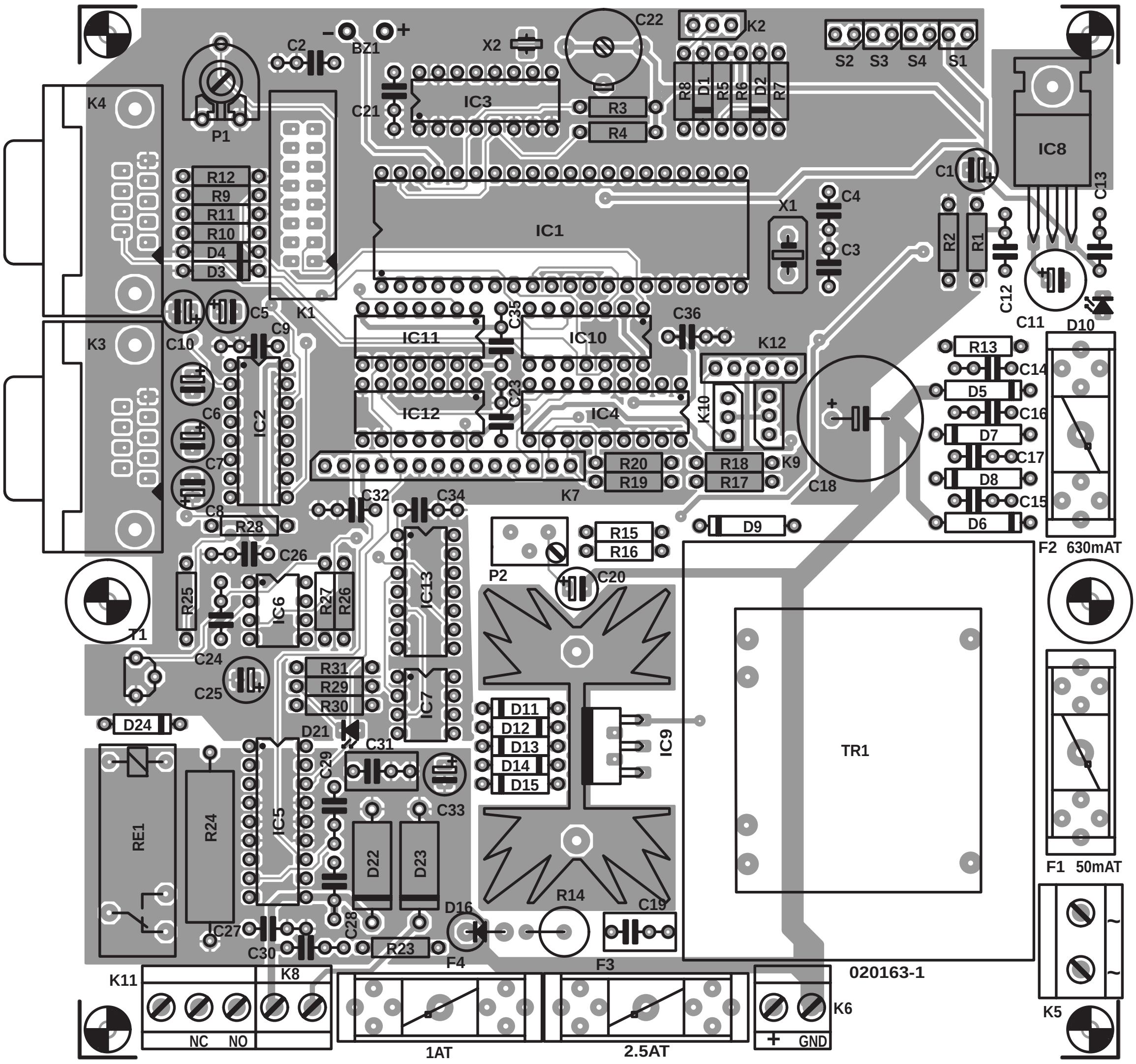


Figure 4. The circuit is assembled on a double-sided printed circuit board. The LCD is connected to K1.

tery charging voltage as small as possible, smoothing capacitor C18 has been generously dimensioned. D16 prevents current from flowing in the reverse direction during a mains outage. Diodes D11–D15 are included to provide temperature compensation. If the temperature increases, the voltage across the diodes will decrease, which will cause the charge voltage on the battery to increase. The charge voltage can be adjusted within certain limits using P2.

Components list

Resistors:
R1,R17-R20 = 1kΩ
R2 = 47kΩ
R3,R4,R25,R26,R31 = 10kΩ
R5,R6 = 1kΩ5
R7,R8 = 100Ω
R9-R12 = 27kΩ
R13 = 1kΩ5
R14 = 1Ω5 5W
R15 = 3kΩ9
R16 = 4kΩ7
R21, R22 = not used
R23 = 10Ω
R24 = 2Ω2 5W
R27 = 1MΩ
R28 = 330kΩ
R29 = 220Ω
R30 = 100kΩ
P1 = 10kΩ preset
P2 = 5 kΩ multiturn (vertical mounting)

Capacitors:
C1 = 2µF2/10V radial
C2,C9,C12-C17,C20,C21,C23,C26,C27,C32,C33,C35,C36 = 100nF
C3,C4 = 22pF
C5-C8,C10 = 10µF 16V radial
C11 = 220µF 25V radial
C18 = 2200 µF 35V radial
C19,C31 = 220nF
C22 = 5-40pF
C24,C30,C34 = 22nF
C25 = 100µF 16V radial
C28,C29 = 15nF

Semiconductors:
D1,D2 = 5V1 500mW
D3,D4 = 4V7 500mW
D5-D9 = 1N4007
D10,D18,D20,D21 = LED, red, low current
D11-D15,D24 = 1N4148
D16 = 1N5401
D17,D19 = LED, green, low current
D22,D23 = BYW98
IC1 = AT89S8252-12PC, programmed, order code **020163-41**
IC2 = MAX232
IC3 = PCF8573 P
IC4 = 2803 or TD62083AP
IC5 = L6202
IC6,IC7 = NE555
IC8 = LM7805 TO220
IC9 = L200
IC10 = 74HCT126
IC11 = 74HCT04
IC12 = 74HCT08
IC13 = 1x 74HC00
T1 = BC547B

Miscellaneous:
BZ1 = DC (active) buzzer, 5 or 6V
F1 = fuse, 50mAT (slow) with PCB mount holder
F2 = fuse, 630mAT (slow) with PCB mount holder
F3 = fuse, 2A5T (slow) with PCB mount holder
F4 = fuse, 1AT (slow) with PCB mount holder
K1 = 16-way boxheader
K2,K9,K10 =3-way SIL-header
K3,K4 = 9-way sub-D socket (female), PCB mount, angled pins
K5 = 2-way PCB terminal block, lead pitch 7.5mm
K6,K8 = 2-way PCB terminal block, lead pitch 5mm
K7 = 14-way SIL pinheader
K11 = 3-way PCB terminal block, lead pitch 5mm
RE1 = V23057-B0002-A201 (12-V relay, 1 changeover contact)
S1-S4 = pushbutton, 1 make contact
TR1 = mains transformer, 2 x 9V/12VA (e.g., Monacor/Monarch type VTR12209)
X1 = 12MHz quartz crystal
X2 = 32.768kHz quartz crystal
Heatsink for IC9: Fisher type SK129, 38.1mm high, 6.5 K/W
Sealed lead-acid battery 12V/1.2 Ah
LCD module, 2 x 20 characters, HD44780 compatible, e.g., Hitachi LM032 or JHD JM202A
PCB, order code **020163-1** (see Readers Services page)
Disk set (2 pcs), contains source code files and control software, order code **020163-11** or **Free Download**

Also:
Any no. of iButtons type DS1990A
Optional: iButton holders type DS9093A
Reader probe type DS9092 (optionally -L or -T)

Construction, alignment and commissioning

The circuit is built on a double-sided printed circuit board with plated-through holes, which is illustrated in **Figure 4**. Despite the complexity of the schematic diagram, assembling the board should not present any significant problems. Use IC sockets for all ICs and check that the values and polarities of the components are correct before soldering them in place, since it can be difficult to unsolder components from a circuit board with plated-through holes.

After assembling the board, do not immediately fit the ICs in their sockets, and leave the LCD disconnected for the time being.

Turn P2 fully to the left and connect K5 to the mains. Using P2, adjust the voltage on K6 (the connector for the lead-acid battery) to exactly 13.8 V. Then check the 5-V supply voltage at pins 1 and 2 of K1 and at each of the IC sockets, except for IC4 and IC5 (which receive 13.8 V).

Switch off the mains voltage and rotate C22 so the plates of the capacitor are half engaged. Set P1 to its mid-range position.

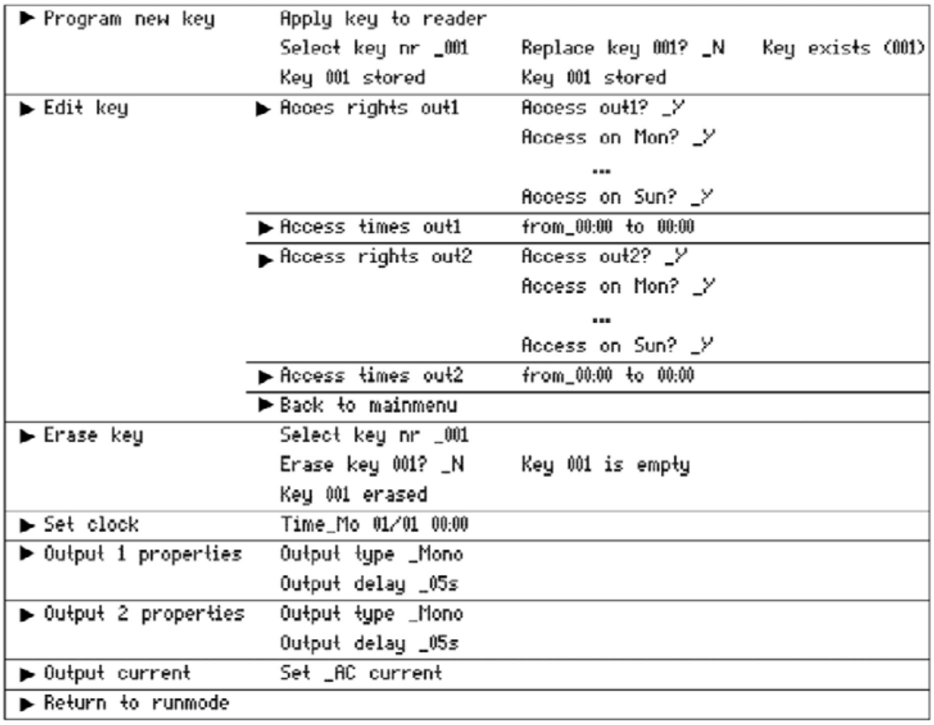
Now insert all ICs into their sockets and connect the LCD to K1 using a length of 16-way flat cable. Switch the mains voltage on again. At this stage, it is quite normal for the circuit to not do anything, since the firmware is missing.

Using a standard 1:1 serial cable, connect the programming port (K4) to a free serial port on a computer. Unzip the Zip file containing the firmware and put all the files in a folder on the hard disk. Start the program MicroFlash.exe, which is located in the folder with the firmware, and select the proper COM port. Click on 'Flash hex' and select file 'iacc-v35.hex' or equivalent (depending on the version number). The microcontroller will now be programmed. Depending on the speed of your computer, this may take a few minutes. As soon as the programming is finished, the microcontroller will be reset and the main program will start to execute in Run mode. Normally, buzzer BZ1 should emit a short beep at this time.

Using P1, adjust the contrast of the LCD. If everything has gone well, the following text will now appear on the LCD: 'iACCESS 3.5 Mo 00:00', 'Run…'

Allow the entire circuit to warm up for 15 minutes, and then use P2 to again adjust the voltage on K6 to 13.8 V as precisely as you can.

If you have a frequency meter, you can connect it between ground and pin 11 of IC3. You can then adjust C22 so the signal has a period of 7.8125 ms (128 Hz). If you don't have



▶ Program new key	Apply key to reader		
	Select key nr _001	Replace key 001? _N	Key exists (001)
	Key 001 stored	Key 001 stored	
▶ Edit key	▶ Acces rights out1	Access out1? _Y	
		Access on Mon? _Y	
		...	
		Access on Sun? _Y	
	▶ Access times out1	from_00:00 to 00:00	
	▶ Access rights out2	Access out2? _Y	
		Access on Mon? _Y	
		...	
		Access on Sun? _Y	
	▶ Access times out2	from_00:00 to 00:00	
	▶ Back to mainmenu		
▶ Erase key	Select key nr _001		
	Erase key 001? _N	Key 001 is empty	
	Key 001 erased		
▶ Set clock	Time_Mo 01/01 00:00		
▶ Output 1 properties	Output type _Mono		
	Output delay _05s		
▶ Output 2 properties	Output type _Mono		
	Output delay _05s		
▶ Output current	Set _AC current		
▶ Return to runmode			

Table 1. Overview of the menu structure.

a frequency meter, you can adjust C22 'by feel', depending on whether the clock runs fast or slow. Even without any adjustment, the clock is quite accurate.

After fitting the circuit board into a suitable enclosure, you should attach an identification label, as illustrated in **Figure 3**. As this is a mains-powered device, this is obligatory in connection with electrical safety.

Basic settings

Provisionally connect four pushbutton switches to the S1–S4 positions on the circuit board. These buttons have the following functions: S1 = Reset, S2 = Enter, S3 = Down, S4 = Up.

Also provisionally connect an iButton probe to pins 1 and 3 of K2, with the centre contact of the probe attached to pin 1 and the outer contact connected to pin 3.

Press and hold the Up button. Briefly press the Reset button and wait approximately two seconds, then release the Up button. The system will enter the configuration mode and display a menu on the LCD. You can now use the Up and Down buttons to page through the various menu options. Do so until the option 'Output 1 properties' appears, and then press Enter. Then select 'Output type _Mono' (you can use the Up and Down buttons to select one of two options: '_Mono' or '_Bi'). Press Enter and select a time, such as 'Output delay _05 s'. Press Up or Down until a time of 5 seconds is set; then press Enter. You will be returned to the main menu. Now select the option 'Output 2 properties', and under this option select 'Output type _Bi' as the output type. Pressing Enter will again return you to the main menu. The door opener interface has thus been configured as a monostable interface with a period of 5 seconds, and the relay output has been configured as bistable.

Select the 'Output current:' option in the main menu and press Enter. Then use Up and Down to select 'Set _AC current' and press Enter.

There is little point in setting the time before the board has been fully fitted in its enclosure and the lead-acid battery has been connected. Select 'Program new key' and press Enter. The message 'Apply key to reader' will appear on the LCD. Hold an iButton against the iButton probe. Once the key has been correctly read, you will hear a short beep, and the physical key number will be shown for 2 seconds on the LCD. After this, 'Select key nr. _001' will appear. After you press Enter, the key will be stored under number 1. If you wish, you can use Up and Down to select a different key number in the range 1–125 before pressing Enter. Key numbers already in use are marked by a # sign (e.g. _065#).

Finally, select 'Return to runmode' in the main menu and press Enter. The system will execute a reset and return to the run mode. Now disconnect the middle lead of the iButton probe and connect it to pin 2 of K2. After this, each time the iButton that has just been programmed into the system is held against the reader probe, relay RE1 should be actuated. You can also test the door opener interface by again connecting the iButton probe to pins 1 and 3 of K2. Connect a load to K8 that can handle a 12-V AC voltage. If you use a 'heavy' load (> 300 mA), you must connect the lead-acid battery to K6 and allow it to charge if necessary. Each time the iButton is held against the reader probe, power will be applied to the load for 5 seconds.

Overview of the menu structure

The menu structure is quite simple and intuitive. If you use **Table 1** as a reference, you should not have any significant problems. Nevertheless, a few remarks are in order with regard to keys and access privileges (which are called 'access rights' in the menus). A particular iButton can only be stored once in the memory. If you try to program the same iButton into the system more than once, an error message will appear.

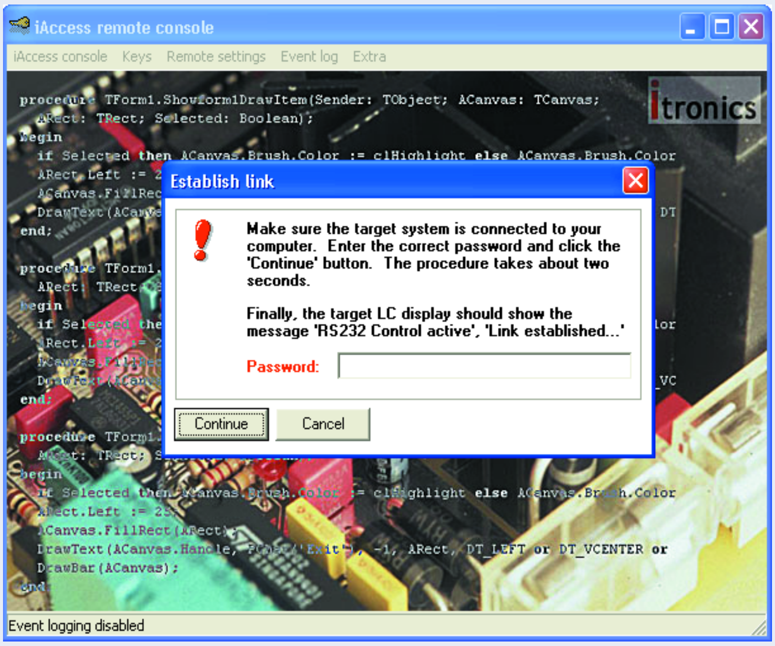
Access privileges are assigned per key at three levels:

– **Level 1:**
door opener or relay output
For each key and each door opener or relay output, you can specify whether the key is allowed to activate the door opener or the relay.

– **Level 2:**
restriction to certain days of the week
For each key and each door opener or relay output, you can specify which days of the week the key is allowed to activate the door opener or relay.

– **Level 3:**
restriction to certain times
An access period can be assigned

Connection to a computer



Make sure the target system is connected to your computer. Enter the correct password and click the 'Continue' button. The procedure takes about two seconds.

Finally, the target LC display should show the message 'RS232 Control active', 'Link established...'

Password:

Event logging disabled

Besides manual operation using the LCD and push-buttons, a computer can also be used to configure all of the system settings. If a large number of keys must be programmed into the system, this method is considerably faster and more convenient than the manual method. This is all made possible by the Windows program 'iAccess remote console', which is also available on the diskettes listed in the Components List.

Using a standard 1:1 serial cable, connect K3 to a free serial port on the computer and then start the program. The first time you use the program, you must check the settings (9600 baud, 8N1) and modify them as necessary.

In the 'iAccess console' menu, select 'Establish link to remote system'. You now have to enter a password (which you can choose yourself the first time you use the program). Click on Continue, and the message 'Link established' will appear after a few seconds. 'RS232 control active' and 'Link established…' will be shown on the LCD. The system is in now in configuration mode.

Now you can use various menus to modify all the configuration settings of the access control system and program or edit keys. It is also possible to generate a backup and/or reset all of the programmed keys, along with their associated access privileges. This function is extremely handy if you wish to update the firmware of a configured system using MicroFlash, since doing so will erase the EEPROM in the microcontroller. When you are finished, select 'Disconnect from remote system' under the 'iAccess console' menu, and the system will return to the run mode.

Each time a key is read, information is displayed on the bottom line of the LCD. This information consists of the time when the key is read, the key reader number, the key number (1–125), and the result code (0–4). The meaning of the result code is as follows:

0 = the presented key is accepted, and the associated door opener or relay output is activated.
1 = unknown key
2 = no access privilege for the door opener or relay in question (Level 1)
3 = no access privilege for the current day of the week (Level 2)
4 = no access privilege outside of the access interval (Level 3)

All of this information can be stored in the computer. This only requires 'Enable event logging' to be activated under the 'iAccess console' menu. If desired, the program can be minimised to an icon in the system tray. It is also possible to start the program in minimised mode, with event logging being immediately active. In addition to the information shown on the LCD, several other items are recorded in the log file, such as:

- result codes 5 and 6 (warm and cold resets);
- the state of the door opener or relay output (on or off);
- the name of the user (see below);
- the physical key number as engraved on the iButton.

If desired, several additional items can be recorded for each key, such as the name of the user (key holder), key label, date issued, date returned, etc. Summaries of this information can also be printed out if desired. This information is also used in the log file and when the keys are edited in configuration mode.

Available software for iAccess

(ref. 020163-11.zip, Free Download)

File name	**Size**	**Description**
Firmware version 3_5.zip	88,139	Microcontroller source code, version 3.5
Executable 2_2.zip	1,087,471	Widows software for recording events and configuring the access control system, version 2.2
Source 2_2.zip	669,005	Source files for the Windows software (Delphi 5). Note: the source code can only be recompiled using Delphi 5 Professional (or a higher version) with the DBISAM database engine installed; see www.elevatesoft.com.

Content of Firmware version 3_5.zip

File name	**Size**	**Description**
IACC-V35.A51	163,172	Version 3.5 source file

IACC-V35.LST	286,299	Listing file generated by the assembler
IACC-V35.OBJ	7,108	Object file
IACC-V35.HEX	19,692	Intel Hex file for programming the microcontroller

Content of Executable 2_2.zip

File name	Size	Description
iConsole.exe	2,300,416	iAccess Remote Console version 2.2 executable file
iconsole.ini	507	Configuration parameters and various settings for iConsole
KEYLOG.DAT	7,807	Database files containing logged event data
KEYLOG.IDX	8,192	
EXTRA.DAT	14,735	Database files containing user data and additional key information
EXTRA.IDX	28,672	
BACKUP.DAT	30,166	Database files for generating and restoring backups of the keys database
BACKUP.IDX	16,384	
INETCTRL.DAT	1,663	Database for receiving external commands via a computer network
INETCTRL.IDX	8,192	

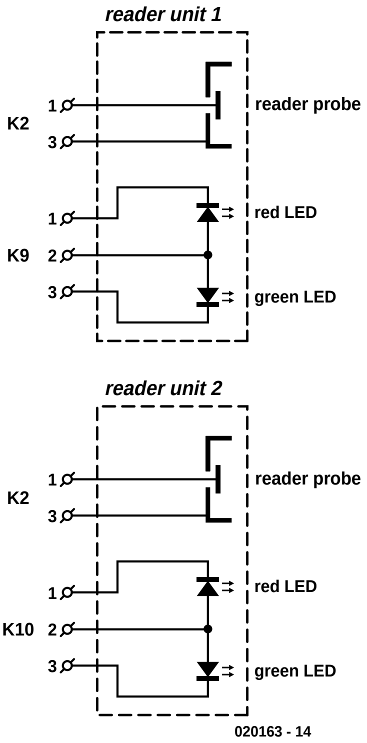


Figure 5. The iButton probes and indicator LEDs (D17–D20) are combined to form two reader units.

Red LED	Green LED	Meaning
blinking at 0.5 Hz	off	system waiting for a key
off	on	door opener or relay activated
on for 3 s	off	key refused
on	off	system in configuration mode

Table 2. Status LED indications.

for each key and each door opener or relay output. The key can activate the door opener or relay output only during this time period.

Note that access privileges at a higher level take priority over access privileges at a lower level. For example, if a key does not provide access to door opener 1 based on Level-1 access privileges, access is effectively denied to the key regardless of what privileges may be assigned at Level 2 or Level 3.

Building and connecting reader units

For practical use, the iButton probes must be fitted into robust reader units that can be installed adjacent to a door or other type of entrance.

For the prototype, cast aluminium boxes were used for this purpose. Besides an iButton probe, each box was fitted with red and green status LEDs. The necessary connections are shown in **Figure 5**.

The wiring between the main circuit board and the reader unit can best be made using standard UTP network cable. Make sure that the terminals of the iButton probe are connected using a single twisted pair in the cable, such as blue and blue/white.

For outdoor applications, you should preferably use a DS9092L probe or possibly a DS9092T probe. The DS9092 model is not suitable for outdoor use, since raindrops can cause shorts between the outer contact ring and the contact surface in the middle. The iButtons can be fitted into small holders (type DS9093) that are intended to be attached to a key ring and to which a number can be affixed.

Finally, **Table 2** provides a brief summary of the indications provided by the status LEDs.

Additional information about iAccess is also available on the author's website at

http://www.itronics-design.be/

(020163-1)

Free Downloads

All project software.
File number: **020163-11.zip**
PCB layout in PDF format.
File number: **020163-1.zip**
www.elektor-electronics.co.uk/dl/dl.htm,
select month of publication.